echo plus
copy program

The smart, easy-to-use program that does away with tedious user-supplied parameters. Copies almost all, even some of the hardest-to-copy Apple II Software.

AGRANAT SYSTEMS

ECHO PLUS USER MANUAL

Copyright 1985 Agranat Systems
All Rights Reserved

ENHANCEMENTS

Agranat Systems reserves the right to make improvements to the software described in this manual at any time and without notice.

DISCLAIMER

Agranat Systems makes no warranties, either express or implied, regarding the ECHO PLUS program or this manual, its merchantability, or its fitness for any particular purpose. In no event will Agranat Systems be liable for direct, indirect, incidental, or consequential damages resulting from the use of or any defect in the ECHO PLUS program or this manual.

REGISTRATION

In order to be eligible for any additional services from Agranat Systems pertaining to this product, you must be a registered owner. To register, send the enclosed ECHO PLUS registration card to Agranat Systems.

BACK-UP POLICY

You may purchace one back-up copy of the ECHO PLUS program disk by sending the enclosed ECHO PLUS back-up order form with $15.00 to Agranat Systems.

If the ECHO PLUS program disk is ever damaged, return it to Agranat Systems with $5.00 to cover postage and handling, and we will replace it as soon as possible.

We will accept checks or money-orders payable to Agranat Systems.

TRADEMARKS

Apple is a registered trademark of the Apple Computer Corporation.

ECHO PLUS is a trademark of Agranat Systems.

NOTICE OF COPYRIGHT

The ECHO PLUS program and this manual are registered copyrights of Agranat Systems. All rights are reserved. No part of the ECHO PLUS program or this manual may be duplicated in any form by any means for any purposes, other than the creation of an archival back-up copy, without written permission from Agranat Systems.

You may be held liable for any damages resulting from the duplication and distribution of your copy of the ECHO PLUS program or this manual.

TABLE OF CONTENTS

INTRODUCTION 2

1. GETTING STARTED 4

2. THE ECHO PLUS DISPLAY 4

3. KEYBOARD CONTROLS 10

4. USING ONE OR MORE DISK DRIVES 10

5. COPY DISK NORMAL 12

6. COPY DISK AUTOMATIC 13

7. ANALYZE FORMAT / CHECK DISK DRIVE ALIGNMENT 15

8. EDIT TRACK 20

9. ERASE DISK 26

10. DISK CERTIFY 26

11. DRIVE SPEED 27

12. OTHER FUNCTIONS 29

13. HOW THE DISK DRIVE WORKS 29

14. HOW STANDARD DOS 3.3 DISKS ARE FORMATTED 33

15. HOW ECHO ANALYZES A TRACK 35

16. COPY PROTECTION SCHEMES 36

17. TIPS ON COPYING SOFTWARE 39

INTRODUCTION

On July 10, 1982, Agranat Systems published ECHO 1.0, the first copy program that could distinguish between sync-bytes and non-sync bytes automatically. Thus ECHO 1.0 could copy more software, with fewer parameters, than the other programs that were available.

Over the last three years, the competition has slowly caught on to many of the innovations in ECHO 1.0. But Agranat Systems was working hard to develop new technology that would leave the competion another few years behind. The result of this effort is ECHO PLUS.

Capable of copying almost all software without any parameters, ECHO PLUS is by far the easiest to use and most powerful copy program available. Furthermore, ECHO PLUS is packed with a combination of advanced features not available in other copy programs.

ECHO PLUS was designed with you in mind. We designed the displays and menus to help make ECHO PLUS easy and convenient to use. We also built so much intellegence into ECHO PLUS that you don't need to know about how disks are protected in order to copy them.

We hope that you will be fully satisfied with ECHO PLUS. We look forward to any comments or suggestions, about the ECHO PLUS software or this manual, that you have to offer. Please be sure to send in your registration card so that we can provide you with discounts on any new versions of ECHO that may become available and keep you up to date on new Agranat Systems products.

ACKNOWLEDGEMENTS

I would like to take this opportunity to thank Michael Reese of Murphysboro, Illinois. Without his encouragement, support, and inspiration, ECHO PLUS would never have been written. I hope we shall meet in person someday...

I would also like to thank June Agranat for her professional marketing advice, and Edward Agranat for helping me write and edit this manual. Thank you for being the best parents anyone could have.

Finally, I would like to acknowledge one of the finest software packages available for Apple][computers, the Gutenberg Sr. Word Processor (a registered trademark of Gutenberg Software, Ltd.). This entire manual was printed using Gutenberg and the NEC PC-8023A-C printer (a trademark of Nippon Electric Co., Ltd.).

A WORD ABOUT SOFTWARE PIRACY

Don't! ECHO PLUS may be used to create archival back-up copies of copyrighted software. However, these copies may not be distributed to any other party. To do so would be a violation of the copyright laws of the United States.

HOW TO USE THIS MANUAL

We know that you are anxious to boot up ECHO PLUS and start copying disks. We suggest that you read Sections 1 through 6 before you begin.

Sections 1 through 4 explain the fundamentals of using ECHO PLUS. Sections 5 through 12 explain all of the ECHO PLUS functions available. Sections 13 through 16 explain technical aspects of copy-protection and are recommended reading. Section 17 contains some useful tips on copying software.

Two numbering systems are used throughout this manual. The decimal numbering system (base 10) is what we use every day. The hexadecimal numbering system (base 16) uses sixteen digits instead of ten. They are 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A (= 10), B (= 11), C (= 12), D (= 13), E (= 14), and F (= 15). The digits in a hexadecimal number (from right to left) are the 1's, 16's, 256's, and 4096's instead of the 1's, 10's, 100's, and 1000's used in the decimal numbering system. For example, the hexadecimal number 1C7 represents 7 1's, C (= 12) 16's, and 1 256's. Thus 1C7 in hexadecimal is equal to 7x1 + 12x16 + 1x256 = 7 + 192 + 256 = 455 in decimal.

Tracks are numbered using hexadecimal notation. Whole-tracks, quarter-tracks, half-tracks, and three-quarter tracks are represented by .00, .25, .50, and .75 respectively. For example, track 1E.50 refers to the half track between 1E (hex) and 1F (hex) or between 30 and 31 decimal.

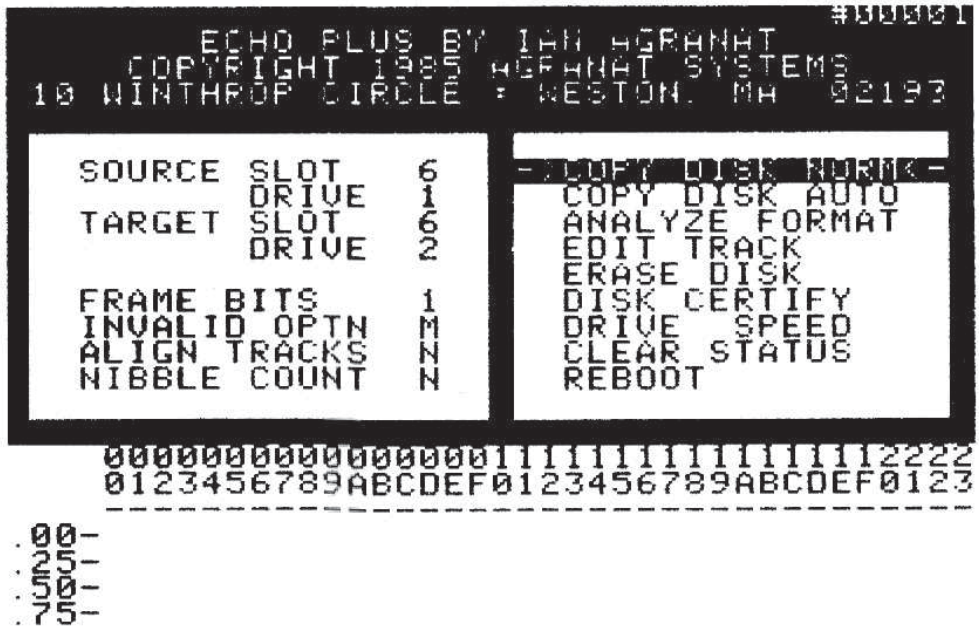
1. GETTING STARTED

1.1. System Requirements

To run ECHO PLUS, you will need an **Apple II, II+, IIe, or IIc** computer system with **48K of RAM.** It is possible to use only one disk drive, but two drives are highly recommended to minimize wear and tear on your original disks as well as your patience! ECHO PLUS should also run on most Apple II compatibles and emulators.

1.2. Booting up ECHO PLUS

Boot up ECHO PLUS as you would boot up a regular DOS 3.3 disk. In less than five seconds, you should see the ECHO PLUS display as shown below. If ECHO PLUS encounters an error while booting, the screen will fill up with inverse '@' characters and ECHO PLUS will attempt to re-boot. If this happens often, your disk drives may be out of alignment. See Section 7.4 for more information.



2. THE ECHO PLUS DISPLAY

The ECHO PLUS display is divided into three windows. On the left hand side of the screen is the Global Parameter Window. This window displays the current values of the eight global parameters used

throughout the ECHO PLUS program. On the right hand side is the Function Window. This window displays menus and function parameters according to whatever mode ECHO PLUS is in. Across the bottom of the screen is the Status Display Window. This window displays information about each track of a disk.

2.1. The Global Parameter Window

The Global Parameter Window shown below lists the eight global parameters and their current values.

```
SOURCE SLOT     6
       DRIVE    1
TARGET SLOT     6
       DRIVE    2

FRAME BITS      1
INVALID OPTN    M
ALIGN TRACKS    N
NIBBLE COUNT    N
```

2.1.1. SOURCE SLOT

The SOURCE SLOT parameter is the slot number of the disk drive controller connected to the source drive. In a standard Apple configuration, the value of this parameter would be six. SOURCE SLOT can have a range from between zero and seven. The default value is the slot from which ECHO PLUS was booted.

2.1.2. SOURCE DRIVE

The SOURCE DRIVE parameter is the drive number from which the original disk will be read. This parameter can have a value of one or two; the default value is one.

2.1.3. TARGET SLOT

The TARGET SLOT parameter is the slot number of the disk drive controller connected to the target drive. In a standard Apple configuration, the value of this parameter would be six. TARGET SLOT can have a range from between zero and seven. The default value is the slot from which ECHO PLUS was booted.

2.1.4. TARGET DRIVE

The TARGET DRIVE parameter is the drive number to which the duplicate copy will be written. This parameter can have a value of one or two. The default value depends on the number of drives found to be present when ECHO PLUS was booted up.

2.1.5. FRAME BITS

The FRAME BITS parameter is the number of extra "zero" bits that are added on to the end of each sync-byte. For more information, refer to Section 13.3. FRAME BITS can range between zero and two. The default value is one.

2.1.6. INVALID OPTION

The INVALID OPTION parameter specifies how ECHO PLUS will handle writing invalid bytes. For more information, refer to Section 13.2. The INVALID OPTION parameter can have the following values:

N	No option - write invalid bytes as is.
M	Modify option - convert invalid bytes into valid bytes before writing.
R	Remove option - remove invalid bytes from track before writing.

The default value of the INVALID OPTION parameter is M for Modify.

2.1.7. ALIGN TRACKS

The ALIGN TRACKS parameter specifies whether or not ECHO PLUS will line up the tracks relative to track zero. For more information, see Section 16.5. This parameter can have the values Y (Yes) or N (No). The default value is N (No).

2.1.8. NIBBLE COUNT

The NIBBLE COUNT parameter determines if and how ECHO

PLUS will preserve the nibble count on a track. For more information, refer to Section 16.4. This parameter can have the following values:

N — No nibble count - do not try to preserve track lengths.

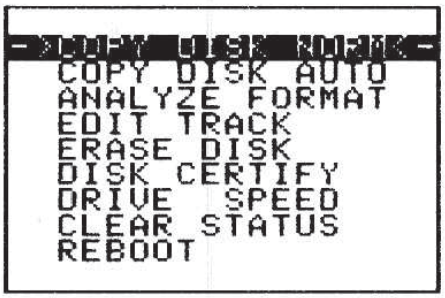
M — Manual nibble count - rewrite the track until the target length is the same as the source length.

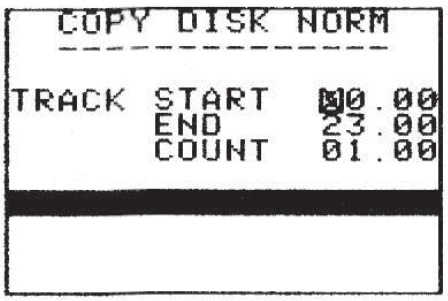
A — Automatic nibble count - rewrite the track until the target length is the same as the source length and adjust the size of the track automatically.

The default value is N (No nibble count).

2.2. The Function Window

When ECHO PLUS is first booted up, the Function Window contains the main menu as shown below left. This is a list of all of ECHO PLUS's functions. When certain functions are selected, the content of the Function Window will be replaced by a list of local parameters that are specific to the given function. For example, if the COPY DISK NORMAL function is selected, the Function Window is replaced by parameters describing which tracks ECHO PLUS will copy as shown below right.





2.3. The Status Display Window

The status display as shown below contains four rows of thirty-six columns. Each column represents one of the thirty six tracks

from 00 through 23 hexadecimal (0 through 35 decimal). The rows divide the tracks into whole, quarter, half, and three-quarter tracks. ECHO PLUS can present one character of information for each quarter track on the disk.

2.3.1. Inverse video characters indicate that ECHO PLUS is currently performing some operation on a track. These characters are decoded as follows:

L ECHO PLUS is Lining up the track relative to the start of track zero.

R ECHO PLUS is Reading from the track.

A ECHO PLUS is Analyzing the track.

W ECHO PLUS is Writing to the track.

V ECHO PLUS is Verifying what it just wrote to the track.

2.3.2. Normal video characters are used to display status about work done on a particular track. These characters are decoded as follows:

L ERROR: ECHO PLUS was unable to Line up the track relative to track zero.

A ERROR: ECHO PLUS was unable to Analyze the track.

W ERROR: The target disk is Write protected.

V	ERROR: ECHO PLUS was unable to Verify what it just wrote to the track.
B	The track is Blank.
C	The track may need to be nibble counted.
*	The track seems to contain valid data.
+	The track may or may not contain valid data.
.	The track probably doesn't contain valid data.
0	The track was copied successfully.

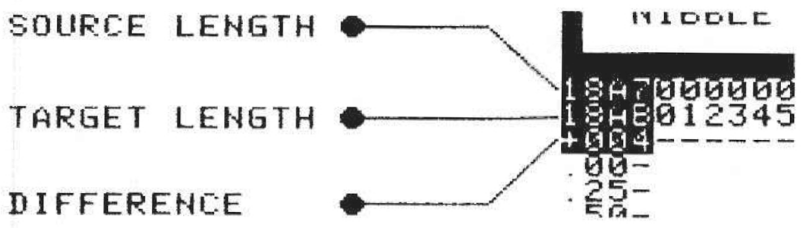
For more information on '*', '+', and '.', refer to Section 7.2.

2.3.3. Track Length Display

There is a small space under the Global Parameter Window and at the upper left corner of the Status Display Window which is used to display information about the length of the track currently being processed. This information is displayed in three lines in the format shown below:

```
XXXX
YYYY
SZZZ
```

where XXXX is the length in bytes of the original track, YYYY is the length in bytes of the copied track, and ZZZ is the difference between the two. S is either '+' if the copied track is longer than the original track, or '-' if the copied track is shorter. These values are displayed in hexadecimal notation.



3. KEYBOARD CONTROLS

3.1. Global Keyboard Controls

There are two keys which have an effect throughout the ECHO PLUS program.

3.1.1. Pressing [CTRL] [RESET] at any time causes ECHO PLUS to return to the main menu, clear the status display, and reset all of the parameters to their original default values.

3.1.2. Pressing the [ESCAPE] key at almost any time causes ECHO PLUS to return to the main menu. The status display and all of the parameters are unchanged.

3.2. Cursor Controls

The cursor is identified by inverse video. It can either cover a parameter digit or a menu option. When ECHO PLUS is first booted up, the cursor is at the top of the Function Window on the COPY DISK NORMAL menu option.

3.2.1. The keys [→], [↓], or [SPACEBAR] move the cursor forward to the next position.

3.2.2. The keys [←] or [↑] move the cursor backward to the previous position.

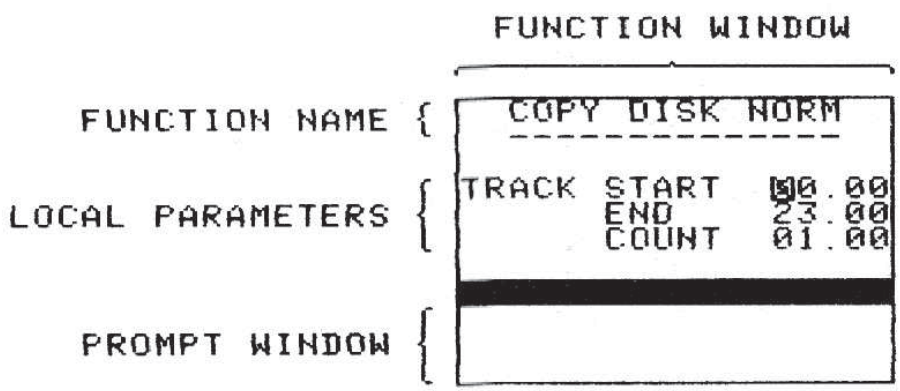
3.2.3. The [RETURN] Key: If the cursor is on a parameter digit, pressing the [RETURN] key moves the cursor forward to the next parameter or menu option. If the cursor is on a menu option, pressing the [RETURN] key causes that option to be selected.

NOTE: Moving the cursor backward from the top of the Function Window places it at the bottom of the Global Parameter Window and vice versa.

4. USING ONE OR MORE DISK DRIVES

When any of the ECHO PLUS functions (except CLEAR STATUS DISPLAY and REBOOT) is selected, the Function Window is divided into two parts as shown below. The upper two-thirds is used to

display the function name and any local parameters or menu options that pertain to the selected function. The lower one-third (called the Prompt Window) is used to tell you when to insert which disks into which drives.



4.1. Getting the prompt

4.1.1. The COPY DISK AUTOMATIC function has no local parameters, so a message is displayed in the Prompt Window as soon as COPY DISK AUTOMATIC is selected.

4.1.2. The DRIVE SPEED function has two menu options for selecting either the source or target drive. As soon as this option is selected, a message will be displayed in the Prompt Window.

4.1.3. For all the other functions, position the cursor on the last local parameter and press the RETURN key. A message will be displayed in the Prompt Window.

4.2. Prompt Window Messages

A Prompt Window message tells you to insert the source or target disks into their proper drives as described below. While ECHO PLUS is waiting for you to do this, the Prompt Window message will be flashing. When you have inserted the disk(s) into their proper drive(s), press the RETURN key and ECHO PLUS will proceed.

4.2.1. INSERT DISKS

If you are using two disk drives and the selected function uses both the original and duplicate disks, this message will be used. Insert the original disk in the source drive and the duplicate disk in the target drive.

4.2.2. INSERT SOURCE

Insert the original disk in the source drive.

4.2.3. INSERT TARGET

Insert the duplicate disk in the target drive.

4.2.4. INSERT BLANK

This message is used with the DRIVE SPEED function. This function may be used with either the source or target drive, but will always write to disk. Insert a blank disk or a disk that you don't mind writing on.

5. COPY DISK NORMAL

This function is used to copy a protected disk when something is known about how it is formatted. COPY DISK NORMAL relies on the global and local parameters to determine how it will copy a given disk. Without changing any parameter values, COPY DISK NORMAL can copy many protected disks, including most business programs.

5.1. Local parameters

The COPY DISK NORMAL function has three local parameters that are used to describe which tracks ECHO PLUS will copy.

5.1.1. TRACK START

The value of the TRACK START parameter indicates the first track that ECHO PLUS will copy. The default value is 00.00.

5.1.2. TRACK END

The value of the TRACK END parameter indicates the highest

numbered track after TRACK START that ECHO PLUS will copy. The default value is 23.00.

5.1.3. TRACK COUNT

The value of the TRACK COUNT parameter is added to the current track to determine the next track to be copied. The default value is 01.00.

5.2. For tips on changing the global and local parameters in order to copy certain kinds of protected disks, see Section 17.

6. COPY DISK AUTOMATIC

The COPY DISK AUTOMATIC function is a unique and powerful feature of ECHO PLUS. Without supplying or changing any parameters at all, COPY DISK AUTOMATIC can copy some of the most difficult-to-copy software.

6.1. How COPY DISK AUTOMATIC works

The COPY DISK AUTOMATIC function begins by analyzing the format of the protected disk, reading and analyzing each quarter track. This procedure takes about ten minutes. For more information about analyzing format, see Section 7.2. When the disk format is analyzed, COPY DISK AUTOMATIC then decides which tracks to copy. At this point, the Status Display Window will be cleared. Then the tracks selected will be indicated by '*' characters in the Status Display Window. Next, each of the selected tracks is read in and analyzed. Some of the global parameters are automatically adjusted and the track is copied. After each track has been copied, the COPY DISK AUTOMATIC mode automatically returns and attempts to re-copy any tracks that resulted in an error. When all tracks selected appear to be successfully copied, ECHO PLUS returns to the main menu.

6.2. Automatic parameter adjustment

6.2.1. The FRAME BITS parameter is set to either one or two, whichever will result in the smallest difference between the original track length and the duplicate track length (assuming a drive speed of about 300 RPM).

6.2.2. The **INVALID OPTION** parameter is unchanged.

6.2.3. The **ALIGN TRACKS** parameter is set to **Y** (Yes).

6.2.4. The **NIBBLE COUNT** parameter is unchanged with one exception. If a track results in a status code 'C' (possible nibble count) and the **NIBBLE COUNT** parameter was **N** (No), then it is changed to **A** (Automatic) for that track.

6.3. Special Commands

While the COPY DISK AUTOMATIC function is analyzing the disk format, there are two special commands that can be issued.

6.3.1. Pressing **SPACEBAR** will cause the COPY DISK AUTOMATIC mode to stop analyzing the disk format and to guess which tracks to copy using incomplete information. This can be used as a shortcut if you think that the disk format is predictable.

6.3.2. Pressing the **RETURN** key causes COPY DISK AUTOMATIC to stop analyzing the disk format and begin copying only tracks that have an 'L', 'A', 'W', 'V', '*', '+', or '.' character in the Status Display Window. This may be useful if, for example, the COPY DISK NORMAL function was used and generated errors on a few tracks. The COPY DISK AUTOMATIC function in this way can automatically re-copy those tracks over again until the errors are corrected.

6.4. COPY DISK AUTOMATIC limitations

There are reasons for not using the COPY DISK AUTOMATIC mode which should be considered. First, and most apparent, the COPY DISK AUTOMATIC function takes a long time to run (typically twenty to thirty minutes). Second, since the COPY DISK AUTOMATIC function makes an educated guess about how to copy each track, it is possible that it would guess incorrectly, and thus be less reliable than COPY DISK NORMAL would be if the proper parameter settings were used. So, the COPY DISK AUTOMATIC mode is best used when you do not know how to copy something, and COPY DISK NORMAL doesn't work with the default parameters. By watching what COPY DISK AUTOMATIC does, you may be able to determine how to set the parameters manually for

COPY DISK NORMAL to work.

7. ANALYZE FORMAT / CHECK DISK DRIVE ALIGNMENT

The ANALYZE FORMAT function is most often used to determine which tracks of a disk to copy, but it can also be used to make sure that your disk drives are properly aligned.

7.1. Local Parameters

The ANALYZE FORMAT function has three local parameters that are used to define the range of tracks that will be analyzed.

7.1.1. TRACK START

The value of the TRACK START parameter indicates the first track that ECHO PLUS will analyze. The default value is 00.00.

7.1.2. TRACK END

The value of the TRACK END parameter indicates the highest numbered track after TRACK START that ECHO PLUS will analyze. The default value is 23.75.

7.1.3. TRACK COUNT

The value of the TRACK COUNT parameter is added to the current track to determine the next track to be copied. The default value is 00.25. Note that this value should not be changed if you want meaningful data about disk format or drive alignment.

7.2. How ANALYZE FORMAT works

The ANALYZE FORMAT function will read in and analyze each track in the range specified by the local parameters. The strength of the magnetic signal recorded on the track is measured statistically. This measurement is displayed in the Status Display Window as either '*' (a strong signal), '+' (a medium signal), or '.' (a weak signal). By observing the pattern created over the whole disk, it is possible to deduce which tracks should be copied if a few simple rules are followed:

7.2.1. Track Zero

If the disk can be booted, there must be at least some information on track 00.00. Therefore, this track should almost always be copied.

7.2.2. Minimum track spacing

Although most disk drives are capable of accessing each quarter track of a disk, they can not effectively record information on all of these tracks. This is because the read/write head (much like a head of a tape recorder) has a certain physical size. If data were written on track 12.00, and then more data were written on track 12.50, the data on track 12.50 would overwrite the data on track 12.00. As a general rule, data should not be less than one full track apart, and can not be less than 00.75 tracks apart.

7.2.3. Signal Strength

A strong signal (indicated by a '*' character in the Status Display Window) will almost always be found on tracks containing valid data. The adjacent quarter tracks will often have a strong signal also. A medium to weak signal will usually be found on tracks half way between tracks containing valid data.

7.2.4. Other Status Codes

If the Status Display Window shows that a track is blank ('B') then it is blank. Furthermore, valid data will not be found less than three quarters of a track away from a blank track. Other kinds of errors usualy indicate that the magnetic signal strength could not be determined.

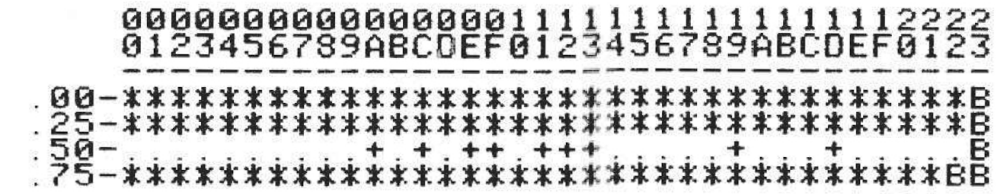
7.3. Examples

Different types of disk format, as shown in the Status Display Window, are described in three examples below.

7.3.1. Example A

After running the ANALYZE FORMAT function on a disk, the

Status Display Window looks like this:

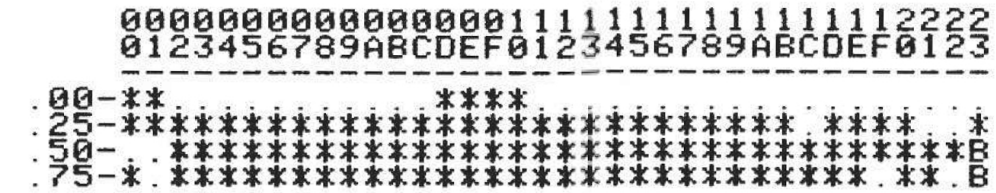


First notice that tracks 22.75 through 23.75 are blank ('B'). This means that the last track with data on it can not be higher than track 22.00 (because a track three quarters of a track away is blank). Also notice that for the other tracks, each whole track and quarter track has a strong signal ('*'), and the half tracks have weak to medium signals ('.' and '+'). Knowing that valid data is usually spaced one full track apart, the center of the strong magnetic signal appears to be on each whole track from 00.00 through 22.00. Thus, the disk in this example has valid data on the whole tracks 00.00 through 22.00 and track 23.00 is blank.

The disk used in this example was the original System Master disk that came with Apple's Disk]I[disk drive. A large percentage of protected software uses this format.

7.3.2. Example B

After running the ANALYZE FORMAT function on a disk, the Status Display Window looks like this:

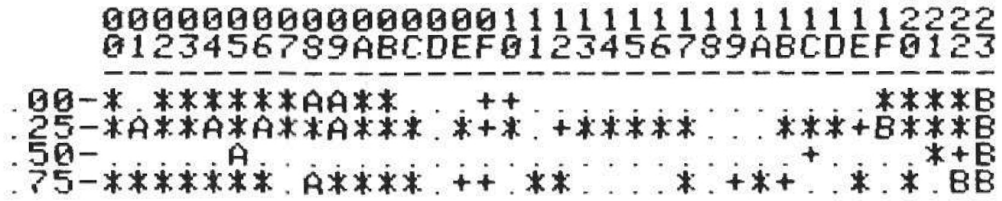


First of all, track 00.00 should have valid data because any bootable disk must access this track. You can confirm this because track 00.00 and adjacent track 00.25 have a strong signal, and track 00.50 has a weak signal. Since tracks containing valid data should not be less than one full track away (and can not be less than three-quarters away), the next

track that could contain valid data would be track 01.00. Since track 01.00 and adjacent quarter tracks 00.75 and 01.25 have a strong signal, and adjacent half tracks 00.50 and 01.50 have weak signals, track 01.00 probably does contain valid data. The next track that could contain valid data would be track 02.00. But this track has a weak signal, so it probably does not. Notice that tracks 02.25, 02.50, and 02.75 have a strong signal and track 03.00 has a weak signal. This indicates that track 02.50 probably contains valid data. By following this logic across the disk, it appears that whole tracks 00.00, 01.00, and half tracks 02.50 through 22.50 contain valid data, and track 23.50 is blank. For this particular disk, the analysis is correct. You may wonder about the deviation to this pattern on tracks 0D through 10 and 1F through 22. Remember that this is a statistical analysis. There will be deviations. Chances are, if ANALYZE FORMAT were run again on the same disk, some of these deviations will go away and others may appear elsewhere. It is possible that tracks 0D through 10 may have a slightly altered format, but it is not likely.

7.3.3. Example C

After running the ANALYZE FORMAT function on a disk, the Status Display Window looks like this:



For now, lets just look at tracks 00 through 0B. Where there are analyze errors (indicated by 'A'), assume that you don't know anything about the signal strength. The whole tracks and quarter tracks appear to have strong signals, and the half tracks appear to have weak signals, so the tracks with valid data are probably the whole tracks from 00.00 through 0B.00.

Now look at tracks 0C through 1E. It seems that the quarter tracks have stronger signals than the whole or half tracks, but that all the signals are relatively weak. This is an example of track arcing. For more information about this kind of protection

scheme, refer to Section 16.6. The way this disk was recorded, there is valid data on both the whole and the half tracks from 0C.00/0C.50 through 1E.00/1E.50. But as mentioned earlier, this is impossible! There can't be valid data less than three-quarters of a track away! However, the way track arcing works, the data only covers a small arc (or several arcs) on a track, and data on adjacent half tracks is recorded in different arcs such that the actual data is a full track away, but adjacent half tracks are used. Since approximately half of each whole and half track is not recorded, the track as a whole shows a weak signal. But the quarter tracks are close enough to both the half and whole tracks on either side that the signal is stronger. Thus the one-quarter and three-quarter tracks have a relatively strong signal, and the whole and half tracks have a relatively weak signal. As it turns out, this kind of protection can usually be copied by copying either the one-quarter or the three-quarter tracks. In this example, the one-quarter tracks seem to have a stronger signal than the three-quarter tracks, so one-quarter tracks 0C.25 through 1E.25 should be copied.

Track 1F and track 23 appear to be blank.

In tracks 20 through 22, there is a strong signal on the whole tracks and the one-quarter tracks. It would seem that the whole tracks 20.00 through 22.00 are recorded.

Thus, to copy this disk, whole tracks 00.00 through 0B.00, quarter tracks 0C.25 through 1E.25, and whole tracks 20.00 through 22.00 should be copied.

7.4. Checking disk drive alignment

Drive alignment refers to the ability of a disk drive to position the read/write head accurately. If the computer tells the disk drive to read track 7, you don't want it to read track 7 1/8! If your drives are out of alignment, you may have a great deal of trouble trying to duplicate copy-protected software even though the drive may seem to function properly.

We strongly recommend that you check the alignment of your disk drive(s) on a regular basis. If your drive is out of alignment, you should take it to an authorized dealer and have them "align and tune" your disk drive(s) for you.

If you run the ANALYZE FORMAT function on an original System Master disk, you should get a pattern resembling the one in Example A above. If your disk drive is out of alignment, the magnetic signal strength will be weaker than it should be on the whole tracks, and may be shifted toward one of the adjacent quarter tracks. Notice that it is important to use the original System Master disk. A copy of the System Master will compare the alignment of your disk drive to the one that made the copy. It is probably safe to assume that the disk drives used to mass produce the original System Master disks for Apple are well aligned.

The display below shows another kind of alignment problem which is less serious, but should be corrected. Tracks 14.00 through 16.00 are properly aligned since the quarter tracks on either side have about the same signal strength (medium), and the whole tracks have a strong signal. But look at tracks 00.00 through 13.00. It seems that the three-quarter tracks have a slightly stronger signal than the one-quarter tracks. This means that for the first half of the disk, the drive is out of alignment (reading slightly forward of where it should). Now look at tracks 17.00 through 22.00. It seems that the one-quarter tracks have a slightly stronger signal than the three-quarter tracks. This means that for the second half of the disk, the drive is out of alignment (reading slightly backward of where it should). Because alignment is correct at about the middle, the drive is properly aligned, but the tracking is not linear. This can cause some problems when trying to copy software, but is not at all as serious as a disk drive out of alignment.

```
    0000000000000000111111111111111122222
    0123456789ABCDEF0123456789ABCDEF0123
    ------------------------------------
.00-**********************************B
.25-++++++++++.++.++++++++**********+B
.50-      .A.A+A                    .B
.75-+*++++*+**********+.+++.++..+A+A.+BB
```

8. EDIT TRACK

The EDIT TRACK function is intended to be used by experienced users of ECHO PLUS. In order to use this powerful track editor effectively, a knowledge of how disks are formatted and protected is needed. Using the EDIT TRACK function, you can read in and examine

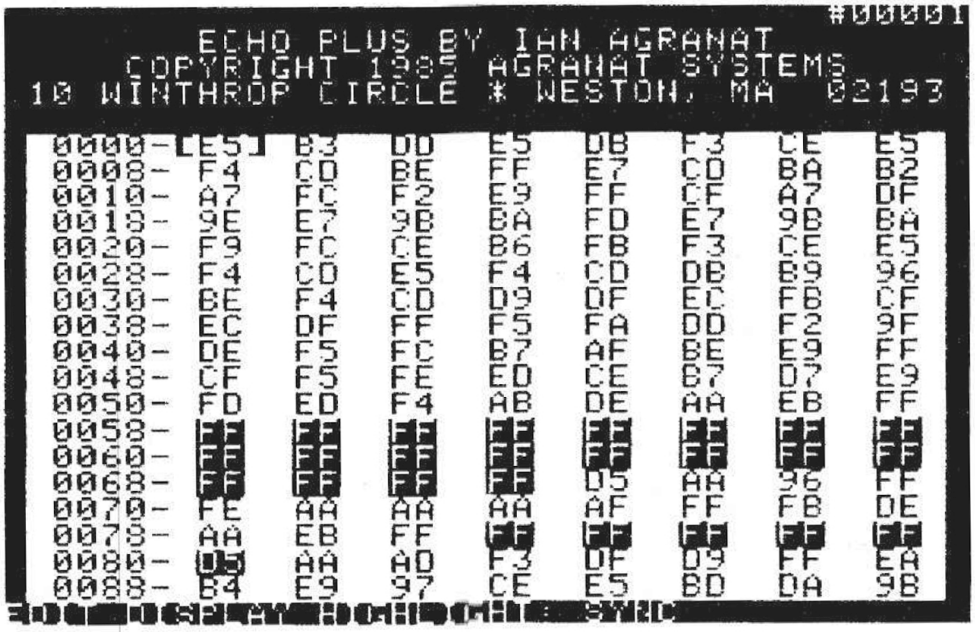
the raw data on a track, analyze the track "by hand", and copy the track under your direct control. This allows you to copy almost everything that can be copied, but may require a great deal of effort, time, and skill.

8.1. Local parameters

The single local parameter specifies which track to edit. The default value is **00.00**.

8.2. The EDIT TRACK display

When the editor is invoked, the specified track is read into the 28K track buffer and the screen format changes to resemble the display shown below.


#000051
ECHO PLUS BY IAN AGRANAT
COPYRIGHT 1985 AGRANAT SYSTEMS
10 WINTHROP CIRCLE * WESTON, MA 02193
0000- E5 B3 DD E5 DB F3 CE E5
0008- F4 CD BE FF E7 CD BA B2
0010- A7 FC F2 E9 FF CF A7 DF
0018- 9E E7 9B BA FD E7 9B BA
0020- F9 FC CE B6 FB F3 CE E5
0028- F4 CD E5 F4 CD DB B9 96
0030- BE F4 CD D9 DF EC FB CF
0038- EC DF FF F5 FA DD F2 9F
0040- DE F5 FC B7 AF BE E9 FF
0048- CF F5 FE E0 CE B7 D7 E9
0050- FD ED F4 AB DE AA EB FF
0058-
0060-
0068- FF FF FF FF D5 AA 96 FF
0070- FE AA AA AA AF FF FB DE
0078- AA EB FF
0080- 05 AA AD F3 DF D9 FF EA
0088- B4 E9 97 CE E5 BD DA 9B

Notice that there are eighteen lines of data with the following format:

XXXX - AA BB CC DD EE FF GG HH

were XXXX is the hexadecimal address of the data AA in the track buffer (from **0000** through 6FFF). BB is at address XXXX + 1, CC is at address XXXX + 2, ... HH is at address XXXX + 7. Thus, each full screen has eighteen lines times eight bytes per line for

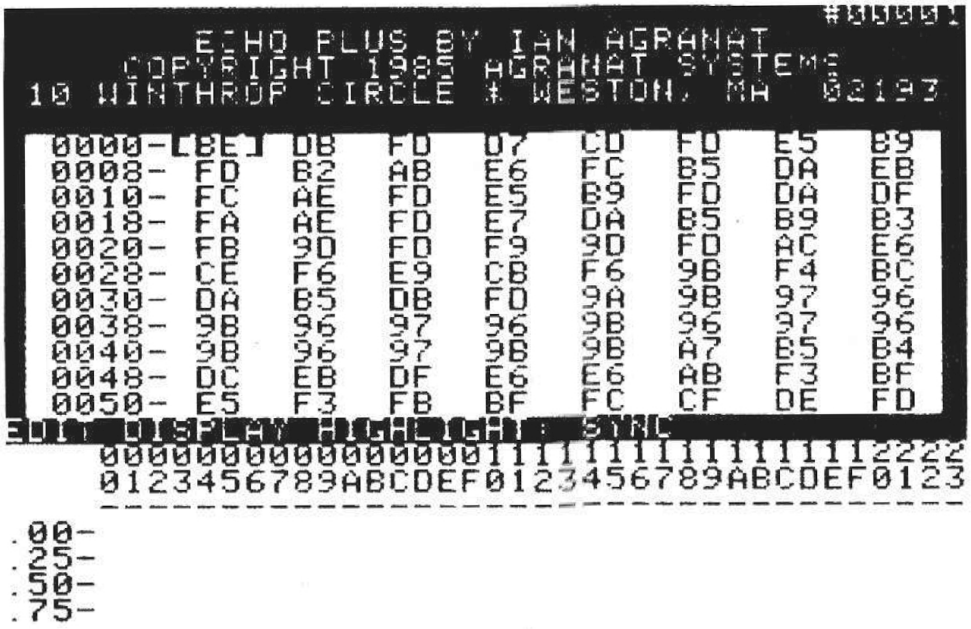
a total of one hundred and forty-four bytes of raw disk data per screen. These bytes should be read left to right and top to bottom. In other words, the first byte read in is stored at address 0000 and the last byte read in is stored at address 6FFF. Note that since the average track length is about 6K bytes long, the 28K track buffer will contain several disk images.

8.3. Edit Display Highlight

The inverse line at the bottom of the EDIT TRACK screen indicates the Edit Display Highlight mode. These modes determine which bytes of data will be highlighted in inverse video. The default Edit Display Highlight is SYNC. This means that sync bytes will be displayed as inverse video and non-sync bytes will be displayed as normal video. The Edit Display Highlight can also be set to INVALIDS (invalid bytes are displayed in inverse video), BOTH (both invalid bytes and sync bytes are displayed in inverse video), or NONE (no bytes are displayed in inverse video). Pressing the [D] key will change the Edit Display Highlight from SYNC to INVALIDS to BOTH to NONE and back to SYNC.

8.4. Split Screen Mode

When the track editor is first invoked, the display is in the full-screen mode. By pressing the [/] key, the display mode can be toggled between full-screen and split-screen modes. In split-screen mode, the Status Display Window is shown at the bottom of the screen as shown below, and the track editor display is cut down to only eleven lines instead of eighteen lines. Note that in full-screen mode, the Status Display Window will appear for a moment while ECHO PLUS is reading, analyzing, writing, or verifying information on the disk.


#00001
ECHO PLUS BY IAN AGRANAT
COPYRIGHT 1985 AGRANAT SYSTEMS
10 WINTHROP CIRCLE * WESTON, MA 02193
0000-[BE] DB FD D7 CD FD E5 B9
0008- FD B2 AB E6 FC B5 DA EB
0010- FC AE FD E5 B9 FD DA DF
0018- FA AE FD E7 DA B5 B9 B3
0020- FB 9D FD F9 9D FD AC E6
0028- CE F6 E9 CB F6 9B F4 BC
0030- DA B5 DB FD 9A 9B 97 96
0038- 9B 96 97 96 9B 96 97 96
0040- 9B 96 97 9B 9B A7 B5 B4
0048- DC EB DF E6 E6 AB F3 BF
0050- E5 F3 FB BF FC CF DE FD
EDIT DISPLAY HIGHLIGHT SYNC
0000000000000000111111111111111122222
0123456789ABCDEF0123456789ABCDEF0123
.00-
.25-
.50-
.75-

8.5. The EDIT TRACK cursor

The cursor in the track editor points to a byte of data and is indicated by enclosing that byte with square brackets. For example:

```
1458 - FF  FF [D5] AA  96  FF  FE  AA
```

shows the cursor on the byte at address 145A in the track buffer which happens to be a D5.

8.6. Cursor Movement

8.6.1. Pressing the [<] key causes the cursor to jump to the beginning of the track buffer (at address 0000 hex).

8.6.2. Pressing the [>] key causes the cursor to jump to the end of the track buffer (at address 6FFF hex).

8.6.3. Pressing the [,] key causes the cursor to jump back one screen.

8.6.4. Pressing the [.] key causes the cursor to jump forward one screen.

8.6.5. The keys [→], [K] or [SPACEBAR] move the cursor forward one byte.

8.6.6. The keys [←] or [J] move the cursor backward one byte.

8.6.7. The keys [↑] or [I] move the cursor up one line.

8.6.8. The keys [↓] or [M] move the cursor down one line.

8.6.9. Pressing the [RETURN] key moves the cursor to the beginning of the next line if Change Mode is off (see Section 8.8.1). If Change Mode is on, the [RETURN] turns Change Mode off, but does not move the cursor.

8.7. Setting Markers

A unique feature of the ECHO PLUS track editor is the ability to set as many as sixty-four pairs of markers anywhere in the data. Markers are displayed by '(' and ')' parentheses. A marker can be toggled in and out of existence by pressing either the [(] key or the [)] key. If you tell the track editor to write the track back to disk, only the bytes between pairs of markers will be written. Furthermore, if there are more than one pair of markers, the space between pairs will be preserved on the track. In other words, track-arcing protection schemes can be copied exactly (something no other copy program can do) by placing markers around the arcs to be copied. Usually, only one pair of markers would be used. The first marker indicates the beginning of the track and the second marker indicates the end of the track. When [CTRL]>[W] is typed, the data between the markers will be Written to the track using the global parameter settings.

8.8. Changing the value of bytes

8.8.1. Change Mode

Pressing the [C] key causes EDIT TRACK to enter the Change Mode. In Change Mode, the cursor brackets are shown in inverse video. This mode allows you to change the value of a byte by typing the hexadecimal digits [0] through [9] and [A] through [F]. Notice that the commands [C] (enter Change Mode), [D] (change Edit Display Highlight), and [F] (Find byte) do not work while Change Mode is activated. Pressing the

[RETURN] key will exit the Change Mode.

8.8.2. Pressing [CTRL] [C] Changes all bytes between markers to the value of the byte under the cursor.

8.9. Changing bytes to sync or normal

8.9.1. Pressing [S] changes the byte under the cursor into a Sync-byte.

8.9.2. Pressing [N] changes the byte under the cursor into a Normal byte.

8.9.3. Pressing [CTRL] [S] changes all bytes between markers to Sync-bytes.

8.9.4. Pressing [CTRL] [N] changes all bytes between markers to Normal bytes.

8.10. Search for byte

Pressing the [F] key moves the cursor to the next occurrence of the byte it was on, or to the end of the buffer if no match was found.

8.11. Track Processing

8.11.1. Pressing [CTRL] [R] causes the track to be Re-read. The cursor is not moved, but the track buffer contents will be replaced. Also, any marks set will be cleared.

8.11.2. Pressing [CTRL] [W] causes the track to be Written to the target drive. When written, the display will show the EDIT TRACK Function Window.

8.11.3. Pressing [CTRL] [A] causes the track to be Analyzed using the ECHO PLUS analyze algorithm. The Status Display Window will show 'A' (Analyze failed), 'B' (track is Blank), 'C' (track may be nibble Count) or a space (no error) according to the result of the algorithm. If the analyze routine succeeds, markers will be set at the start and end of the track.

8.12. Quit

Pressing the [ESCAPE] key quits the track editor and returns to the EDIT TRACK Function Window. Pressing [ESCAPE] a second time will return to the main menu.

9. ERASE DISK

The ERASE DISK function allows you to erase selected parts of a disk by filling it with any repeating one-byte pattern.

9.1. Local parameters

The ERASE DISK function has four local parameters that are used to describe which tracks ECHO PLUS will erase, and how to erase them.

9.1.1. TRACK START

The value of the TRACK START parameter indicates the first track that ECHO PLUS will erase. The default value is 00.00.

9.1.2. TRACK END

The value of the TRACK END parameter indicates the highest numbered track after TRACK START that ECHO PLUS will erase. The default value is 23.00.

9.1.3. TRACK COUNT

The value of the TRACK COUNT parameter is added to the current track to determine the next track to be erased. The default value is 01.00.

9.1.4. ERASE BYTE

The value of the ERASE BYTE parameter is written repeatedly all over each track to be erased. The default value is 00.

10. DISK CERTIFY

The DISK CERTIFY function is used to check the physical surface of a disk for damage (e.g. scratches).

CAUTION

This function will destroy data
on the diskette being certified.

10.1. Local parameters

The DISK CERTIFY function has three local parameters that are used to describe which tracks ECHO PLUS will certify.

10.1.1. TRACK START

The TRACK START parameter indicates the first track that ECHO PLUS will certify. The default value is 00.00.

10.1.2. TRACK END

The TRACK END parameter indicates the highest numbered track after TRACK START that ECHO PLUS will certify. The default value is 23.00.

10.1.3. TRACK COUNT

The value of the TRACK COUNT parameter is added to the current track to determine the next track to be certified. The default value is 01.00.

10.2. How DISK CERTIFY works

Each track selected by the local parameters is first written with a special pattern which is then verified. If the pattern could not be Verified (indicated by a 'V' in the Status Display Window), then the diskette probably has been physically damaged and should not be used. It is also possible that your disk drive(s) are not working properly.

11. DRIVE SPEED

The DRIVE SPEED function allows you to check and adjust the speed of your disk drives. It is sometimes necessary to change the disk drive speed of one or two drives in order to successfully copy some software.

CAUTION
This function will destroy data on
the disk used, so use a blank disk!

NOTE
Adjusting the disk drive speed yourself
may void the warranty on your disk drive.

11.1. Selecting which drive

The DRIVE SPEED function does not have any local parameters. Instead there are two menu options. Choosing the first one allows you to measure the drive speed of the source drive, and choosing the second one allows you to measure the drive speed of the target drive. When you have selected which drive to check, the Prompt Window will flash the message 'INSERT BLANK'. Remember that this function will write to the disk you insert into the specified drive, so use a blank disk.

11.2. How DRIVE SPEED works

Special data is written on track 00.00 of the blank disk and is then read back. The speed of the disk drive in RPM (Revolutions Per Minute) is calculated and displayed. The normal disk drive speed is 300.00 RPM. As a general rule, it is better (in terms of copying software) for a disk drive to run slightly slow rather than fast.

11.3. DRIVE SPEED display

The speed of the drive is displayed in the Function Window in the format

```
DRIVE=XXX.XX RPM

NORM =300.00 RPM
```

where XXX.XX is the speed of the selected drive in Revolutions Per Minute. If the DRIVE SPEED function can't determine the drive speed because the disk is not seated properly, the drive speed will be displayed as ??????.

12. OTHER FUNCTIONS

12.1. CLEAR STATUS DISPLAY

Selecting this function clears the contents of the Status Display Window and returns to the main menu.

12.2. REBOOT

Selecting the REBOOT function causes the default source slot to be booted.

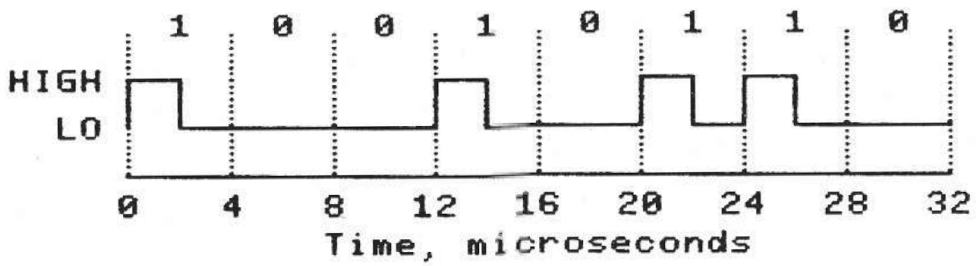
13. HOW THE DISK DRIVE WORKS

In order to make back-up copies of copy-protected software, it is helpful to know a little about software protection. To understand software protection schemes, you must understand how data is stored and retrieved on a disk using a disk drive.

13.1. Storing data on a disk

13.1.1. Bits stored as magnetic fields

Data is stored to and retrieved from the surface of a floppy disk much the same as music is recorded to and played back from the surface of a cassette tape. The diagram shown below shows how digital data (in the form of a string of zero-bits and one-bits) is converted into an electrical signal. This electrical signal is then recorded on the disk surface in the form of a magnetic field. Notice that each bit is recorded as a magnetic pulse which lasts for four microseconds. Each one-bit is converted into a signal which is 'high' for the first half and 'low' for the second half. Each zero-bit is converted into a signal which is 'low' for the whole four microseconds.



As you can see, it is easy to identify one-bits when the magnetic field is read back because each one-bit has a rise and fall in magnetic field magnitude. On the other hand, zero-bits are harder to identify. If there are several zero-bits in a row, the only way of telling how many there are is to see for how long the magnetic field magnitude is low. This presents an interesting problem:

Because the disk drive speed varies from drive to drive, and even varies on one drive as it spins, it is difficult to time signals accurately. For example, one disk drive may write ten zero-bits on a track recorded as ten times four (forty) microseconds of low magnitude magnetic field. If a second disk drive, running ten percent slower, tries to read the ten zeros, it would find forty-four microseconds of low magnitude. The second disk drive would think that there were eleven bits and not ten!

Disk drives are designed to assume that no more than two zero-bits will occur in a row. It is possible to write more than two zero-bits in a row on a track, but they can not be read back reliably. In fact, the disk drive hardware tends to generate one-bits when a low magnitude magnetic field persists for too long, so reading more than two zero-bits in a row can have unpredictable results.

13.1.2. Grouping bits into bytes

When reading data from the disk drive, the one-bits and zero-bits are grouped into eight-bit bytes. The hardware assumes that the left-most bit (called the high bit) of each byte has the value one. For example, assume the following stream of bits is read:

0 1 0 0 1 0 1 1 0 1 1 0 0 1 0 1 1 0 0 1 1 1 1 0 1 0 1

Because the high bit must be a one and the first bit read in is a zero, the zero is skipped over. The second bit is a one, so it will be the high bit of the first byte 1 0 0 1 0 1 1 0 which translates to the hexadecimal 96. The next bit is a one, so it will be the high bit of the second byte 1 1 0 0 1 0 1 1 which translates to the hexadecimal CB. The next two bits are zero-bits, so they are skipped. The next one-bit is the high bit of

the third byte 11110101 which translates to the hexadecimal F5. This grouping of the above stream of bits is shown below.

0(10010110)(11001011)00(11110101)

13.2. Invalid Bytes

As described above, reading data with more than two zero-bits in a row will have unpredictable results. Therefore, any data-byte on the disk with more than two zero-bits in a row is unreliable. An invalid byte is any byte on the disk with more than two zero-bits in a row. If an invalid byte is read, it is possible that bits may have been inserted or dropped. The INVALID OPTION parameter of ECHO PLUS can be used to filter these invalid bytes. Using the Modify option, invalid bytes are changed into bytes that are not invalid before they are written to the target disk. Using the Remove option, all invalid bytes found on a track are deleted (thus shortening the track length) before the track is written. If No option is specified ('N') then invalid bytes are left unchanged.

13.3. Sync Bytes

Because tracks are circular, they have no beginning and no end. If you start to read a track, you will read a stream of one-bits and zero-bits. Since these bits have to be grouped into eight-bit bytes, there must be some way to tell which bit begins a byte. This is why the hardware requires that the high bit of each byte be a one. A sync-byte is a special kind of byte. It consists of a valid eight-bit byte (usually hexadecimal FF, but some other values can be used as well) followed by one or two extra zero-bits. Thus, a sync-byte is nine or ten bits long. These extra bits are called Frame Bits. Notice that the FRAME BITS parameter of ECHO PLUS determines how many frame bits to write on the end of each sync byte.

When several sync-bytes are found in a row (at least four with two frame bits or eight with one frame bit), they are called a sync-byte field. These sync-byte fields are what guarantee that the data bits are grouped into bytes correctly. No matter which bit of the byte the drive starts reading in the beginning of a sync-byte field, it will be reading the bytes correctly when it starts reading the data following the sync-byte field as illustrated in the following eight examples:

a) (11111111)00(11111111)00(11111111)00(11111111)00(11111111)

b) 1(11111110)0(11111111)00(11111111)00(11111111)00(11111111)

c) 11(11111100)(11111111)00(11111111)00(11111111)00(11111111)

d) 111(11111001)(11111110)0(11111111)00(11111111)00(11111111)

e) 1111(11110011)(11111100)(11111111)00(11111111)00(11111111)

f) 11111(11100111)(11111001)(11111110)0(11111111)00(11111111)

g) 111111(11001111)(11110011)(11111100)(11111111)00(11111111)

h) 1111111(10011111)(11100111)(11111001)(11111110)0(11111111)

13.4. Address fields

When a track is read, there must be a way to determine where the data starts. This is done with what is called an address field. An address field is a series of bytes unique to a given track which precedes the data. On a sectored disk format such as DOS 3.3., each sector has a corresponding address field. The address field often contains information such as a track number, sector number, and disk volume number. In order for the address field to be read properly, it must be preceded with a sync-byte field.

13.5. Data encoding

You may be wondering how you can possibly store all possible combinations of data on a disk when each byte must begin with a one-bit and can not have more than two zero-bits in a row. With these restrictions, only 81 (decimal) distinct bytes can be used and there are 256 (decimal) possible bytes. The way this is done is to use some form of encoding where each byte of data would be coded into more than one byte on the disk. There are an endless number of ways to do this. One of the simplest is illustrated below. In this example, each data byte is encoded into two disk bytes. The first of the disk bytes contains four of the bits in the original data byte, and the second disk byte contains the other four bits of the original data byte. By placing one-bits between each data bit, there can never be more than one zero-bit in a row.

Original data:	a b c d e f g h
First disk byte:	1 a 1 c 1 e 1 g
Second disk byte:	1 b 1 d 1 f 1 h

13.6. Checksumming

Disk drives are not reliable. A disk may be seated improperly, the drive may be out of alignment, the disk surface may be damaged, etc. It is important that the data read in be as reliable as possible by checking it for errors. Checksumming is a technique of error checking. There are many checksumming techniques available. Perhaps the easiest to understand is an addition checksum. All the bytes of data are added together and the sum is encoded and stored on the disk with the data. When the data is read back, each byte of data is added up and the sum is compared with the correct sum on the disk. If the two sums disagree, then there is something wrong with the data. By performing a checksum on data, it is possible to realize that corrupted data was read, and the software reading the disk can try again.

13.7. Tracking

The read/write head is moved from track to track with a four-phase stepper motor. There are thirty-six tracks on a disk numbered from the outside inward. The outermost track is 0 (hex) and the innermost track is 23 (hex). Each track corresponds to two stepper motor phases, so the read/write head can move one half track at a time. Therefore, the read/write head can directly access not just the thirty-six whole tracks, but also each half track in between. Quarter tracks are reached by turning on two phases of the stepper motor at once. This causes the read/write head to move between phases, and thus to the one-quarter and three-quarter tracks.

14. HOW STANDARD DOS 3.3 DISKS ARE FORMATTED

In order to better understand how protected disks are formatted, it is helpful to know how unprotected disks are formatted.

14.1. Tracks used by standard DOS 3.3

Standard DOS 3.3 uses the whole tracks from 00.00 through 22.00 to store data. Track 23.00 is not used.

14.2. Sectoring

There are sixteen sectors on each track of a DOS 3.3 disk. Each sector starts with an address field that identifies the disk volume, track, and sector. The address field is followed by a data field which contains the data stored in the sector. Both of these fields are preceded by a sync-field. When a DOS 3.3 disk is initialized (with the DOS command INIT HELLO, for example), the address, data, and sync fields are initially written on the disk. Later, when data is written to a sector, only the data field (and the preceding sync-field) is written. It is important that each address and data field be preceded by a sync-field if individual sectors will be written. Otherwise, writing a sector will cause the address field of the next sector to be read incorrectly.

14.3. Sync-fields

The standard DOS 3.3 sync-fields consist of several FF (hex) sync-bytes in a row.

14.4. The DOS 3.3 Address Field

The address field of a DOS 3.3 disk is fourteen bytes long and has the following format:

D5 AA 96 vv vv tt tt ss ss cc cc DE AA ED

The first three bytes, D5 AA 96, make up a header that identifies the field as an address field. The disk volume, track, sector, and checksum are recorded in two bytes each using the encoding scheme described in Section 13.5. The checksum is calculated by exclusive-or'ing (similar to adding) the disk volume, track, and sector values together. The last three bytes of the address field, DE AA ED, make up a trailer that identifies the end of the address field.

14.5. The DOS 3.3 Data Field

The data field of a DOS 3.3 sector is 348 (decimal) bytes long. The first three bytes are D5 AA AD and identify the field as a data field. The last three bytes are DE AA ED and mark the end of the data field. The other 342 (decimal) bytes are decoded into 256 (decimal) bytes of data and a checksum.

14.6. Standard DOS 3.3 Data Encoding

The method of encoding used by DOS 3.3 is very complex. Recall that there are eighty-one valid bytes that can be stored on a disk. Sixty-four of these are used to encode the data. With sixty-four distinct values, six bits of information can be encoded into one disk byte. For every three data bytes, there are twenty-four data bits. These bits are divided into four six-bit bytes. Thus DOS 3.3 encodes every three bytes of data into four bytes on the disk. The details of this process are beyond the scope of this manual.

15. HOW ECHO ANALYZES A TRACK

Many people wonder why you can't just read data from one disk, write it to another disk, and expect to have a working copy. The primary reason that you can't is because the disk drive speed varies, so the number of bits that can fit on a track varies also. If you write a track full of data to a faster disk drive, the disk will make one complete revolution before all the data is written. This means that the rest of the data will write over the beginning of the track, known as "overwriting". On the other hand, if you write a track full of data to a slower disk drive, there will be a gap between where you stopped writing and where you started writing, known as "underwriting". If a track is overwritten or underwritten in the middle of an address or data field, then the copy will not boot.

The data on each track of an original disk is intentionally underwritten, because it is not likely to fit perfectly on a track, and would obviously not work if it were overwritten. Therefore there is a space between the end and the beginning of the data on each track of an original disk.

In order to make a working copy of a track, it is important that the inevitable overwriting or underwriting occurs in the underwritten space on the original track. In other words, the track should ideally

be written starting and ending at the beginning of this underwritten space. Thus, to copy a disk, it is neccessary to identify this space on the original track; this is the tough part!

Since any valid data on a track must be preceded by sync-bytes, then there must be sync-bytes following the underwritten space on a track. The underwritten space itself usually contains either invalid bytes or sync bytes.

Earlier copy programs used pattern recognition (guided by large numbers of complex sets of parameters) to identify sync-bytes and determine where to break the track. This meant that different parameters were needed to copy almost every disk. ECHO 1.0, the predecessor of ECHO PLUS, was the first copy program ever published that could distinguish between sync-bytes and non-sync bytes automatically. This technology enabled ECHO 1.0 to copy a wide range of software without requiring lists of parameters.

ECHO PLUS can identify sync-bytes with even greater precision than ECHO 1.0. Furthermore, the ECHO PLUS "Analyze" routine is very effective in determining exactly where to break the track (by setting markers at the beginning and ending of the track) in order to copy it successfully.

16. COPY PROTECTION SCHEMES

Many methods of copy-protecting software have evolved over the years, and so has the ability of copy programs to copy them. The first forms of copy-protection where nothing more than simple modifications of standard DOS. Some more recent forms of copy-protection are so innovative that some software can not be copied at all without expensive hardware, or very tedious manual effort.

16.1. Modified DOS

One of the earliest forms of copy-protection involved modifying the standard DOS format so that ordinary copy programs (like COPYA on the Apple System Master) would not work. These modifications included things like changing the address field header from D5 AA 96 to something else, changing the checksumming procedure, etc. ECHO PLUS will be able to copy all modifications of standard DOS using COPY DISK NORMAL with the default parameters, or COPY DISK AUTOMATIC.

16.2. "Self-destruct"

A dangerous form of copy-protection involves software that tries to write to a normally write-protected disk. The original disk would come write-protected to prevent it from destroying itself. If a copy is made, you may forget to write-protect the copy and the program will self-destruct when you run it!

CAUTION
Always write-protect your copies
if the original was write-protected!

16.3. Non-standard tracks

Another form of copy-protection involved using tracks that were not used by standard DOS. These included the half-tracks and track 23.00. ECHO PLUS can determine which tracks to copy by using the ANALYZE FORMAT function. The COPY DISK AUTOMATIC function will do this automatically.

16.4. Nibble Counting

Nibble counting takes advantage of the fact that any copy will inevitably (drive speed variation) be overwritten or underwritten. The original track length of a nibble-counted track is compared to the measured length of the track. If the lengths differ by more than one byte (to account for variance on an underwrite of a few bits), then the software will realize that it is a copy and not an original. It is impossible to tell by looking at a track wether or not it uses nibble counting, but some older forms of nibble counting are easy to recognize. In many cases, a nibble-counted track will be filled with a one-byte value (usually FF) followed by a very small pattern. ECHO PLUS will flag any track that looks like this with a 'C' (possible nibble count) character in the Status Display Window after analyzing the track.

There are two ways that the track length can be preserved by using the NIBBLE COUNT parameter of ECHO PLUS.

16.4.1. Automatic nibble counting

In automatic nibble counting mode, ECHO PLUS will write the track to the target drive until the source and target track

lengths match. The length of the track is automatically adjusted. Since sync-bytes are one or two bits longer than normal bytes, the track length can be shortened by converting non-sync bytes into sync-bytes and vice versa. This will put the correct number of bytes on the track, but the result is not necessarily as good as can be obtained by Manual Nibble Counting.

16.4.2. Manual nibble counting

In manual nibble counting mode, ECHO PLUS will keep trying to write the track to the target drive until the source and target track lengths match. You must then manually adjust the speed of your target disk drive to adjust the length of the track.

NOTE
Adjusting the disk drive speed yourself
may void the warranty on your disk drive!

16.4.3. If for some reason (boredom) you want ECHO PLUS to stop trying to nibble count a track, press the [SPACEBAR].

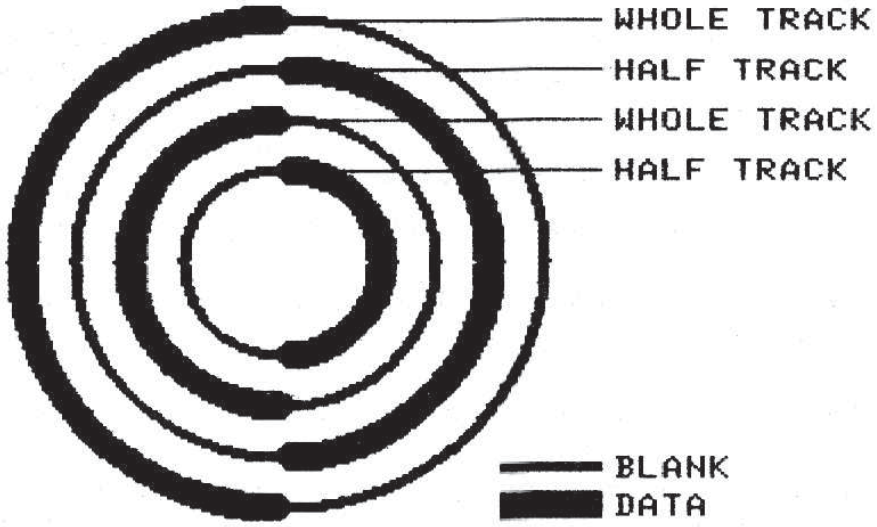
16.5. Aligned tracks (also called synchronized tracks)

This method of copy-protection relies on the angular orientation of data on tracks with respect to each other. The original software checks the orientation of the tracks on the disk and compares this orientation with what it knows to be the correct orientation. If the orientations are different, then the software will realize that it is a copy. This kind of protection is impossible to recognize by looking at a track, but can be copied by setting the ALIGN TRACKS parameter to Y (Yes). The COPY DISK AUTOMATIC function will do this automatically.

16.6. Track Arcing

Because of the physical size of the read/write head, the data on tracks should not be less than one full track apart. Track arcing is a technique that stores data on adjacent half tracks. This is possible because only a part of each track actually contains data, and the tracks are oriented such that the data are one full track apart. The illustration below shows a typical example of track arcing. In this example, each track has data in a 180 degree arc,

and the arcs are arranged so that the data are no less than one full track apart. Some forms of track arcing may have several arcs on a track.



Most copy programs can only copy entire (360 degrees) tracks. Therefore, these copy programs can not copy track arcing protection accurately because when writing one track, the data on the adjacent half tracks is overwritten. The ECHO PLUS track editor is capable of copying tracks with as many as sixty-four arcs of data.

The easiest way to copy track arcing protection is to copy the quarter tracks in between the arcs of data. The quarter tracks are close enough to both adjacent arcs of data that this method often works well.

To recognize track arcing protection, the ANALYZE FORMAT function can be used. See Section 7.3.3 for a track arcing example. COPY DISK AUTOMATIC will often be able to copy track arcing protection without any parameters.

17. TIPS ON COPYING SOFTWARE

17.1. Business software vs entertainment software

Many forms of copy-protection are not reliable. Not all disk drives are capable of booting all forms of copy-protection. For

example, disk drives which are slightly misaligned may not be able to boot software protected by non-standard tracks or track arcing protection. Disk drives which run very fast or very slow may not be able to boot software protected by aligned tracks.

It is important that software oriented toward the business world be reliable. It does not do a software company's reputation any good if their product can't run on some machines. Therefore, most business/professional software is protected using some form of modified DOS. ECHO PLUS will be able to copy most of this software using the default parameters in COPY DISK NORMAL.

There are two kinds of entertainment software: adventure games and arcade games.

Most adventure games access the disk extensively to load various parts of the game into memory in no particular order. For convenience on the programmers part, the protection on adventure games is usually just a modification of standard DOS. Often, aligned tracks or nibble counting may also be used. ECHO PLUS can copy many of the adventure games with COPY DISK NORMAL and the default parameters. If the default parameters do not work, try setting ALIGN TRACKS to Y (Yes). You may also try setting the INVALID OPTION parameter to N (No option).

Most arcade games do not use the disk in such a complex manner as adventure games do, so the protection on arcade games is often more elaborate. Non-standard tracks are almost always used. Track arcing is common among many of the more recent arcade games. Some arcade games are protected so well that they simply cannot be copied using ordinary disk drives. ECHO PLUS can copy most arcade games, without any parameters, using the COPY DISK AUTOMATIC function. By watching what COPY DISK AUTOMATIC does, you can often find a way to copy the disk more accurately using COPY DISK NORMAL.

17.2. What to do when you get errors

A — Analyze error: This usually occur when ECHO PLUS can not determine where to break the track. This can happen if there are only a few sync-bytes on a track and ECHO PLUS could not detect them. By trying over and over again, you will probably be able to get it to work. Try using COPY DISK AUTOMATIC's automatic error retry mode. See Section 6.3.2.

B — Blank: If a track is blank, it is possible that the original program will check to make sure that it is blank. You may want to erase blank tracks using the ERASE DISK function.

C — Possible nibble Count track: Try using automatic or manual nibble counting. See Section 16.4.

L — Can't aLign track: In order to align tracks, ECHO PLUS searches for track zero, sector zero. It assumes that the original disk can be booted using DOS 3.3. In order for ECHO PLUS to align tracks, track 00.00 must be copied first. If ECHO PLUS still can not align tracks, then set the ALIGN TRACKS parameter to N (No).

V — Verify error: This usually means that ECHO PLUS overwrote the track. If you are using two frame-bits, try using only one. If that does not work, try slowing down your disk drive. Setting the INVALID OPTION to R (Remove) will shorten the track by deleting any invalid bytes, so this may also work, but is less desirable because the track will be altered.

17.3. Other tips

17.3.1. Always write-protect your copy if the original disk is write-protected. See Section 16.2.

17.3.2. If you can not figure out which tracks to copy, try copying the quarter tracks 00.25 through 23.25.

17.3.3. If the copy does not work, recopy.

17.3.4. If the copy still does not work, try manual or automatic nibble count. If that does not work, try setting the INVALID OPTION parameter to N (No option). It may also help to reverse the order of disk drives (SOURCE DRIVE = 2, TARGET DRIVE = 1) or to use just one disk drive.

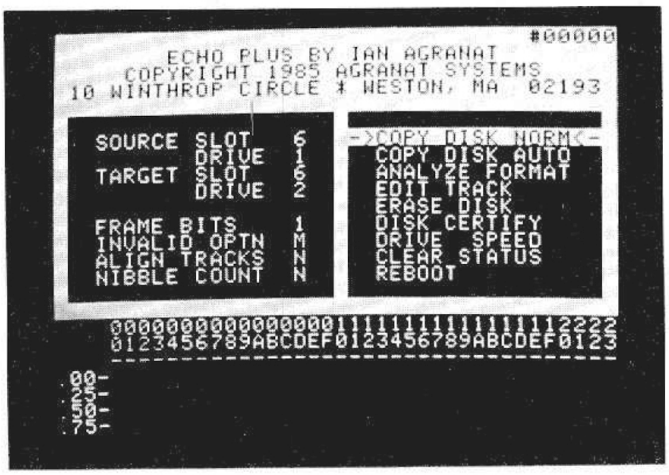
17.3.5. If you can't copy basic DOS software(like the System Master) with ECHO PLUS, chances are that your disk drives are not properly aligned. See Section 7.4.

17.3.6. Remember that some software simply can not be copied. Neither ECHO PLUS nor any other copy program is capable of copying some copy-protected disks. However, very few disks are protected this well.

17.3.7. If all else fails, you can ask Agranat Systems for advice through the mail. Please let us know the title and publisher of the software that you are trying to copy, and how you have tried to copy it. Enclose a stamped, self-addressed envelope, and we will write back to you as soon as possible.

Echo Plus™ is the most advanced copy program available today.
The powerful Echo Plus is the natural outgrowth of the innovative, original Echo 1.0. Now packed with new refinements and sophisticated features. Echo Plus will copy almost all, even some of the best protected software easily and quickly.

Forget about parameters and get right down to copying.
With Echo Plus, you don't have to look up and change as many as 500 parameters as you do with other copy programs. The smart Echo Plus takes care of everything for you.



Designed for humans
In designing Echo Plus, the author gave the human interface aspect as high a priority as the excellence of the program itself. You'll see the difference in the easy-to-read screen format, the cursor-controlled menus, the command flexibility and the thorough, easy-to-follow user manual.

Exclusive Automatic Disk Copier
Now you don't have to give up on those difficult-to-copy disks. Just switch to the exclusive automatic mode. Echo Plus analyzes the format of the protected disk and selects which tracks to copy. Then it reads in and analyzes each selected track, adjusts parameters as necessary, and copies the track.

More pluses
Fast – Echo Plus boots up in only 3.7 seconds and can copy most disks in under two minutes.
Precise drive speed – accurately displays drive speed in RPMs rather than vague and imprecise graphics.
Drive Alignment – checks the alignment of your disk drives.
Disk Certify – checks the surface of a blank disk for physical damage.
Disk Erase – erases selected tracks.
Sophisticated track editor – writes up to 64 separate arcs of data per track to disk. Highlights sync and/or invalid bytes. 28K byte track buffer. Packed with commands.
42-page manual – comprehensive and easy-to-understand. Describes in detail how disk drives work, how disks are formatted, and several copy-protection schemes.

System requirements
For the Apple II, II+, IIc, or IIe computers and most Apple compatibles and emulators with at least 48K RAM and one disk drive (DOS 3.3).

AGRANAT SYSTEMS
10 Winthrop Circle, Weston, MA 02193